

REMARKS/ARGUMENTS

This Amendment is in response to the final Office Action mailed July 16, 2010. Before this Amendment, claims 32-42, 44-48, 50-52, 55-56, 59-60, and 65-68 were pending. In this Amendment, claims 32-33, 44-45, 50, 56, 59, and 65-68 have been amended, claims 46-47, and 55 are canceled, and no new claims are presented for consideration. After entry of this Amendment, which is respectfully requested, claims 32-42, 44-45, 48, and 50-52, 56, 59-60, and 65-68 will be pending. Reconsideration of the rejected claims is hereby requested.

I. EXAMINER INTERVIEW

On September 3, 2010, a telephonic interview was conducted between the undersigned and Examiner Gofman. During the interview, differences between claim 32 and Lee (US 2002/0129024) were discussed. Also discussed were possible amendments to claim 32, although no agreement was reached. Applicants wish to thank the Examiner for his time and careful consideration of the matters presented.

II. CLAIM REJECTIONS UNDER 35 U.S.C. § 103

The Office Action rejected all the pending claims (claims 32-42, 44-48, 50-52, 55-56, 59-60, and 65-68) under 35 U.S.C. § 103(a) as being unpatentable over Lee (US 2002/0129024) (hereinafter “Lee”) in view of Chung et al. (US 6,850,947) (hereinafter “Chung”).

To establish a prima facie case of obviousness, the prior art reference, or references when combined, must teach or suggest all of the claim limitations. There also must be a teaching or suggestion to combine the references or it must be shown that it would have been obvious for one skilled in the art at the time of the invention to try the combination. Applicants respectfully traverse the rejections because the cited references fail to teach or suggest all of the claim limitations and the cited references fail to teach or suggest a motivation to combine the references.

For example, claim 1 recites:

32. A method of transforming data, the method comprising:
positioning a definition pointer to point at a first compound
transform definition within a transform definition file;
invoking a first parallel processing thread to read the pointed at
first compound transform definition;
searching data to be transformed for a data element to be
transformed, the search being responsive to the first compound transform
definition;
**calling a dynamic function defined in the transform definition
file, the dynamic function located elsewhere in the transform definition file
from the definition pointer position;**
transforming any found data element into an output data file,
responsive to the first compound transform definition and called dynamic
function, a data structure of the output data file being responsive to a data
structure of the first compound transform definition;
positioning a definition pointer to point at a second compound
transform definition within the transform definition file;
invoking a second parallel processing thread to read the pointed at
second compound transform definition;
searching data to be transformed for another data element to be
transformed, the search being responsive to the second compound transform
definition; and
transforming any found data element into the output data file,
responsive to the second compound transform definition, the data structure of the
output data file being responsive to the data structure of the second compound
transform definition.

(emphasis added). Independent claims 44, 59, and 65 have been similarly amended.

Lee fails to teach “calling a dynamic function defined in the transform definition file, the dynamic function located elsewhere in the transform definition file from the definition pointer position” as recited. Instead, Lee’s identity server retrieves peripheral programs, such as its navigation bars, from a program service. This program service, such as program service 1660 in FIG. 37 (reproduced below), is *separate* from its XML data registry and files that define the translation, such as elements 1670-76.

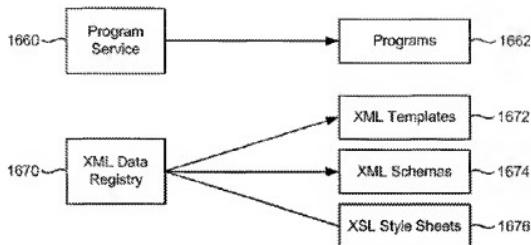


FIG. 37 of Lee (US 2002/0129024 A1)

Lee's peripheral programs are not part of its XML data registry or other files. This is further evident in Lee's example XML files in paragraphs [0305], [0309], [0312], and [0318], which lack a dynamic function as claimed. Therefore, Lee's identity server calling Lee's peripheral programs does not disclose calling a dynamic function "defined in the transform definition file, the dynamic function located elsewhere in the transform definition file from the definition pointer position" as claimed.

Nor is there a teaching or suggestion in Lee of its peripheral programs being defined with its appropriate XML files. Lee's peripheral programs include such programs as navigation bars. A navigation bar lets a user scroll through the text of a response to his request about a certain user (see Lee paragraph [0330]). Such a navigation bar is typically binary code in an executable or DLL (dynamic link library), not something that would be put in its ASCII XML files, let alone an XML template file that defines the output of its translator, such as the example in Lee paragraph [0309]. One skilled in the art at the time of the invention would not have been motivated to put such binary code into Lee's XML files because the otherwise ASCII file would be filled with relatively unreadable binary characters. The binary characters would presumably lack XML tags that separate and classify portions of XML files. It would not have made sense to put navigation functions in the XML files because Lee's XML template is for translating data, not for navigating data.

In an embodiment, a call to a dynamic function within the same file as the simple/compound transform definitions offers flexibility to an end user in some embodiments by allowing highly complex and non-linear, and custom transformations of input data into output data. For example, a dynamic function could determine that employees who are attorneys in California and have last names beginning with H-M and who have not graduated from law school in the last three years are required to attend a minimum continuing learning education seminar. Because the dynamic function is in the transform definition file of the embodiment, an end user could easily edit it so that attorneys in Washington D.C. were included.

Lee's peripheral programs, such as its navigation bars, are not dynamic functions located elsewhere in the transform definition file from the definition pointer position as claimed. Chung is directed toward concurrently reading data organized in non-overlapping data partitions from data stores using parallel processing and fails to cure the deficiencies of Lee.

Because neither Lee nor Chung teach or suggest each limitation in the claims, no combination of the references can render the claims unpatentable under § 103. For at least the above reasons, Applicants respectfully request withdrawal of the rejections of the claims and all claims depending therefrom.

III. AMENDMENTS TO THE CLAIMS

Unless otherwise specified or addressed in the remarks section, amendments to the claims are made for purposes of clarity and are not intended to alter the scope of the claims or limit any equivalents thereof. The amendments are supported by the specification and do not add new matter.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance and an action to that end is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 925-472-5000.

Appl. No. 10/661,167
Amdt. dated September 8, 2010
Amendment under 37 CFR 1.116 Expedited Procedure
Examining Group 2162

PATENT

Respectfully submitted,



Mark Mathison
Reg. No. 57,556

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 925-472-5000
Fax: 415-576-0300
MPM:cta
62860437 v1